Using Combinatorial Optimization to Improve Planning and Scheduling

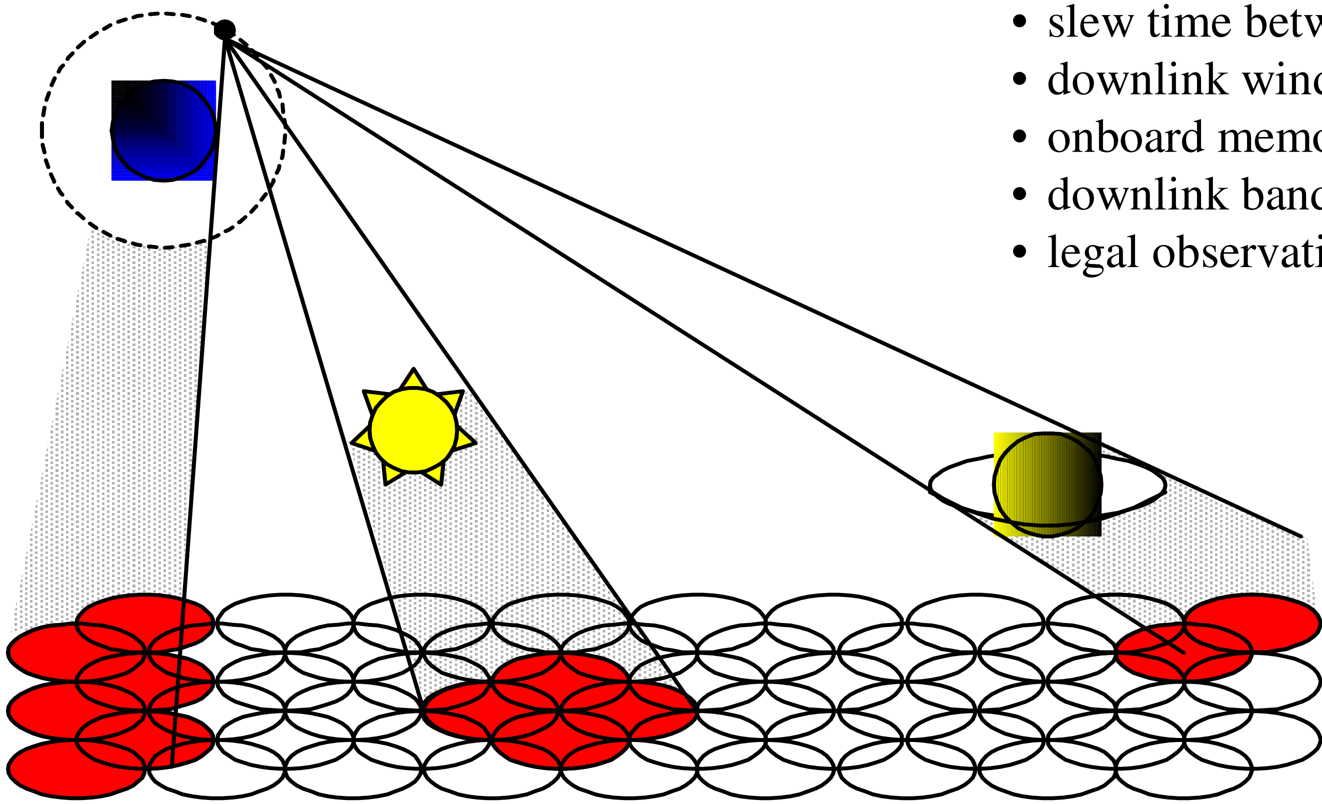
Ben Smith, JPL
Russell Knight, JPL
Richard Korf, UCLA

UCLA

Motivation

- Planning problems can contain optimization sub-problems that interact with the overall problem.
  - TSP (observation scheduling, path planning)
  - Bin packing: downlink scheduling
- Planners are expressive but poor at comb. opt.
  - Can express the full planning problem, but . . .
  - They can’t solve combinatorial optimization problems very well
- Optimization algorithms are powerful but restricted
  - Excellent at solving comb. optimization problems, but . . .
  - They can’t reason about full planning problem
- Need algorithms that can exploit strengths of both.

Example: Observation Scheduling Problem



- Take and downlink observations
- Minimize time, number downlinks
- Satisfy constraints
  - slew time between observations
  - downlink windows
  - onboard memory
  - downlink bandwidth
  - legal observation times

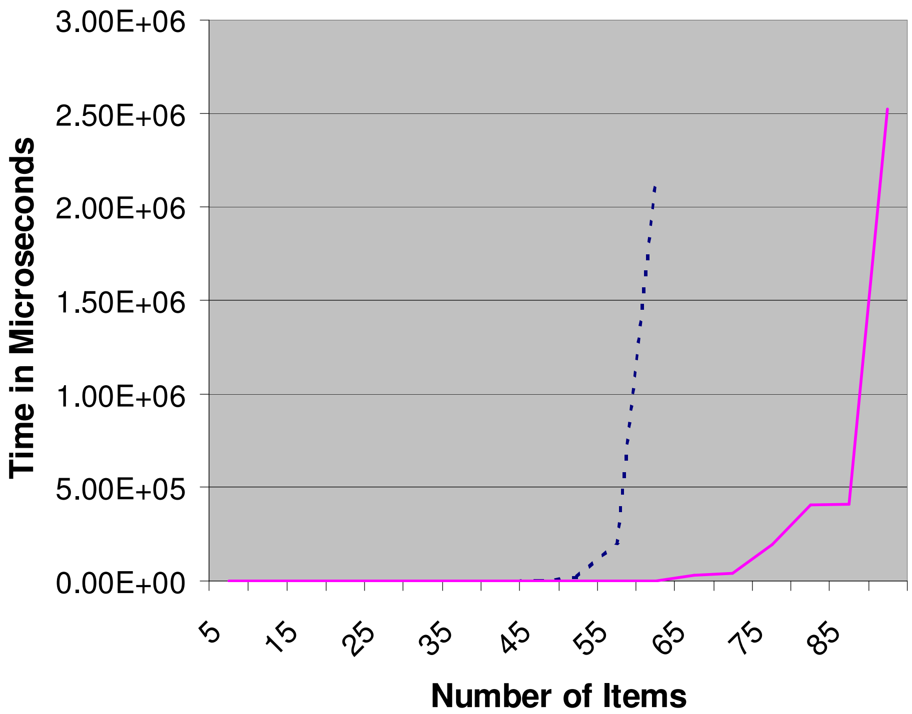
General purpose schedulers do poorly

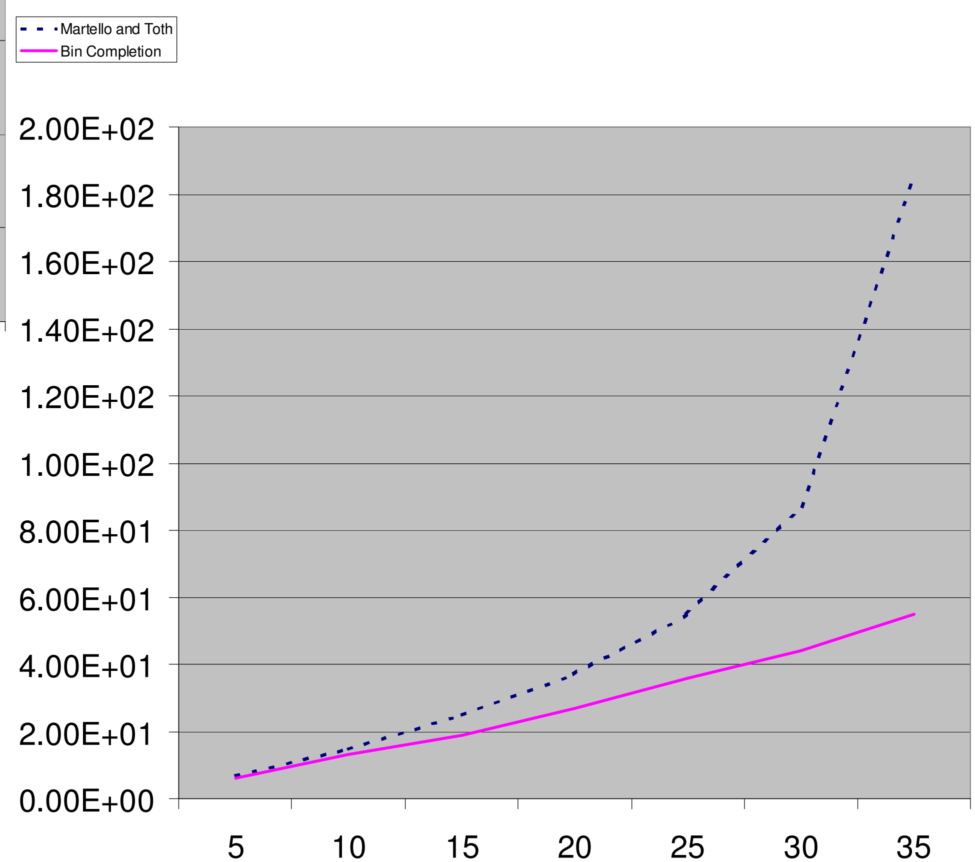
- Planner/scheduler using uninformed iterative repair
  - Can express entire problem
  - Quickly reaches feasible but low quality solution
  - Additional time provides small gains, but soon “maxes out”
- Integer Programming
  - Takes several minutes to reach low quality feasible solution
  - Consistently gains quality with time
  - Yields optimal solutions with enough time (hours to days)
- But can we quickly reach a high quality solution?

Exploit Combinatorial Sub-problems

- TSPTW: Find minimum-makespan observation schedule:
  - City = obervation
  - distance = slew
- Binpack: assign observations to downlinks
  - Bin = downlink opportunity
  - Item = observation
- Have excellent solvers for both of these
  - TSP has many solution algorithms (best depends on TSP properties)
  - Binpack: developed optimal solver under this task
- Need to control interactions
  - TSP solver doesn't consider downlink constraints
  - binpack solver doesn't consider TSP constraints
  - Neither solver considers additional constraints in plan model

Optimal Bin Pack Algorithm vs. Martello & Toth





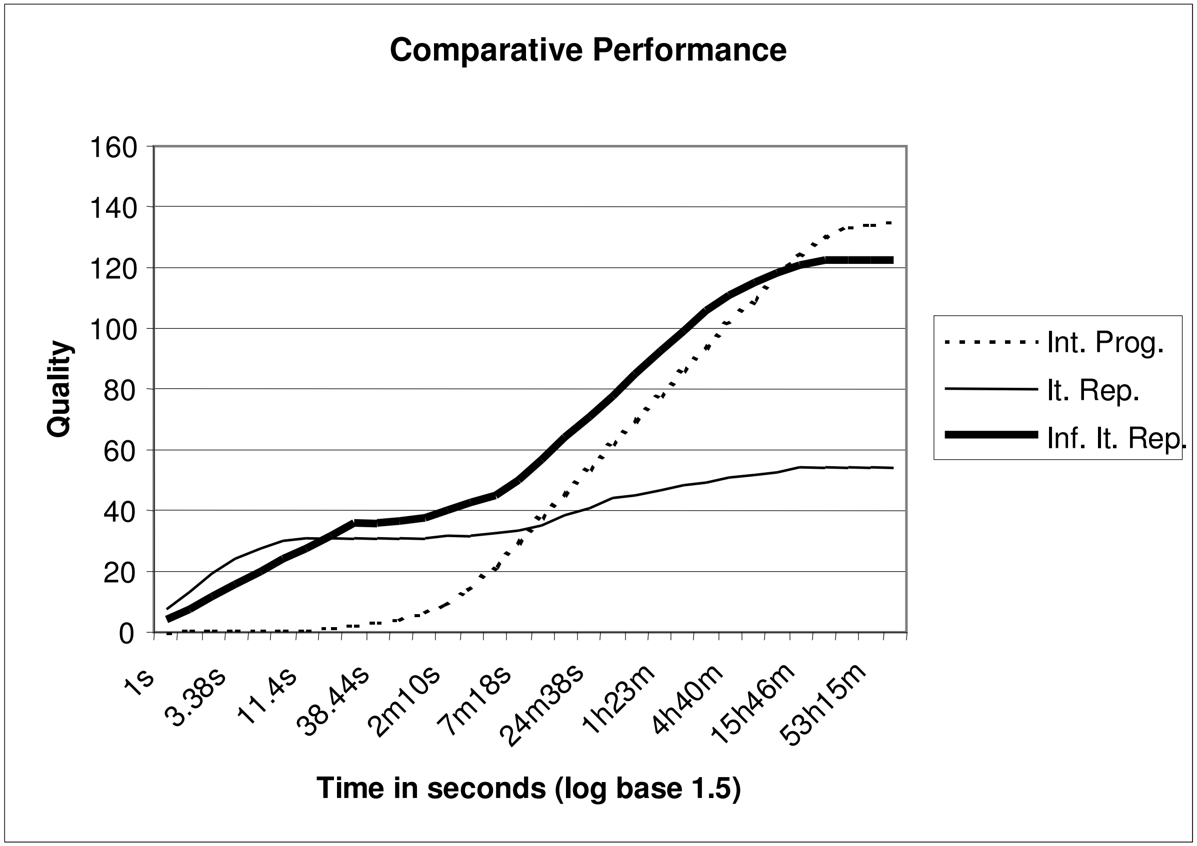
We have developed the best known algorithm for finding optimal bin-packings (Korf AAAI'02). We compare our results with those for the Martello and Toth algorithm.

Iterative Repair approach to combining algorithms

- Break algorithm into independent operators
  - TSPTW: swap with k-opt heuristic; insert edge
  - Binpack: assign item to bin; best-fit first assignment heuristic
- Map planning conflicts to solver operations
  - Maps moves in solver space to moves in plan space
- Specialized Algorithm Criteria
  - Reason in the violated constraint space or limit the application to iterative optimization
  - Reason in small, discrete steps
  - The closer the candidate algorithm matches these criteria, the better it integrates with the planner and other algorithms

Iterative Repair/Integer Programming results

- Iterative repair has best performance over “practical” area of the curve.
- Uninformed iterative repair works well on the very short time scales
- Integer programming performs best only during the impractical area of the curve.

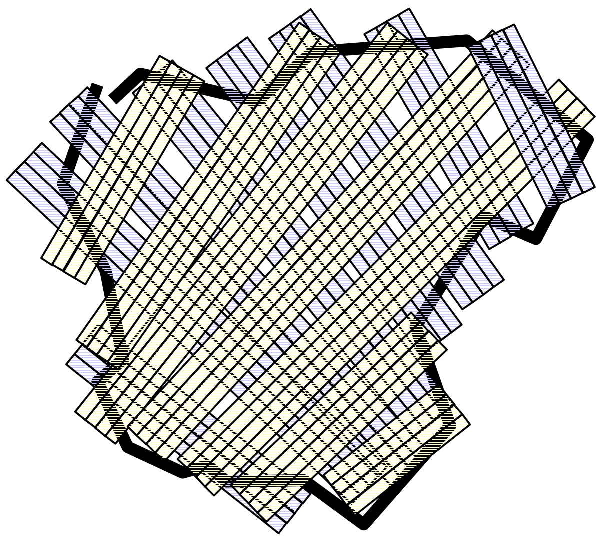


IP Formulation

- TSP Components based on the Grötschel & Holland Formulation
  - Select edges to be included in the solution, select times for each city that respect the edge orderings.
  - Why not use the Dantzig--Fulkerson--Johnson formulation?
    - Ours this is similar, but complete without exponentially many constraints.
  - Why not use the Miller--Tucker--Zemlin formulation?
    - More real variables are required.

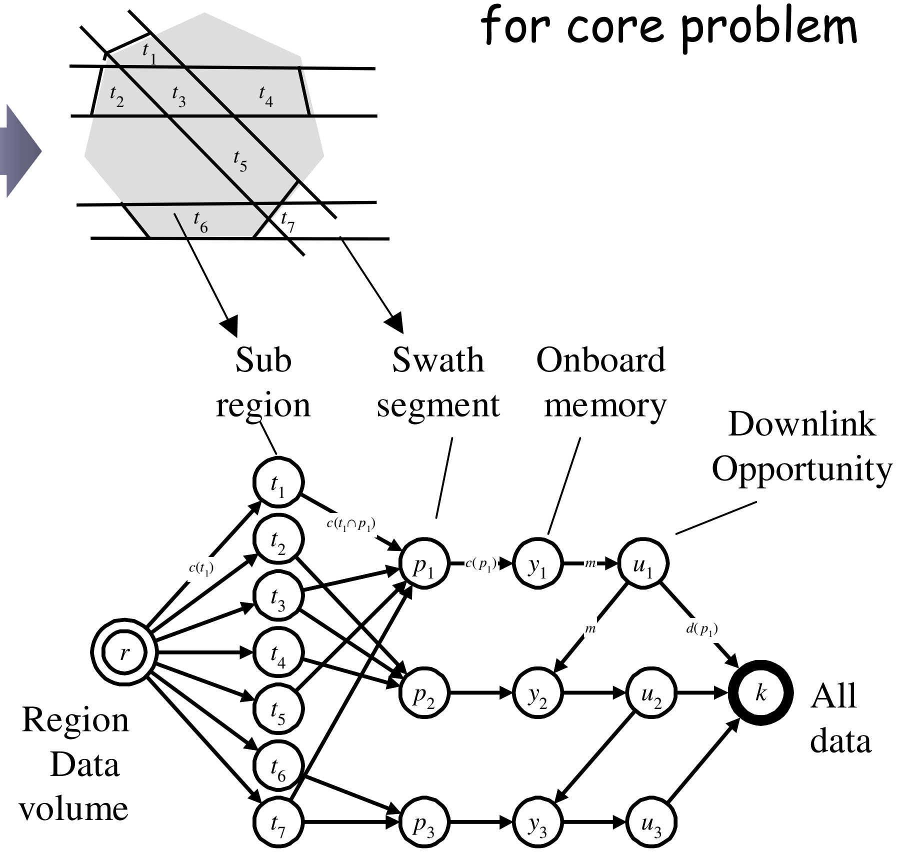
- Bin Pack based on Padberg formulation
  - Start with a maximum number of bins, choose to ignore some and assign values to others.

Swath Selection Problem



- Select **instrument swaths & downlink opportunities** to cover region of interest.
- Mapping schedule interacts with other mission constraints.

Optimization algorithm for core problem

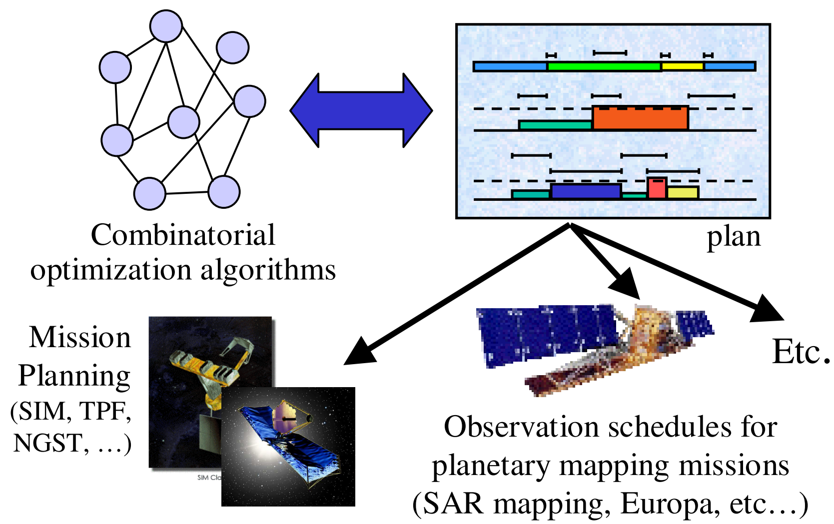


Express as a network flow problem.
Minimize data transport cost through network.

Summary

- 'planner-in-control' interaction algorithm
- Bin-packing solver implemented for solving downlink scheduling sub-problem.
  - New optimal, anytime bin-packer
  - Existing methods do not guarantee optimality
- Swath algorithm formulation and baseline IP and QP formulations

Using Combinatorial Optimization Algorithms to Improve Automated Planning & Scheduling



TASK OBJECTIVES:

Enable planners to solve large, complex planning problems infeasible for current technologies.

Specifically, develop algorithms to solve planning problems comprised of strongly interacting combinatorial optimization sub-problems.

TECHNICAL INNOVATIONS:

Novel coordinated search algorithms that will enable general-purpose planners and combinatorial optimizers to work together effectively.

SCHEDULE:

Milestones	FY01	FY02	FY03
'solver-in-control' algorithms	X		
'planner-in-control' algorithms		X	
specialized solvers for swath scheduling problems		x	X
'local-search' planner algorithms			X

NASA RELEVANCE:

- **Enable planners to solve large, complex problems infeasible for current technologies, such as:**
  - **Optimal planetary mapping schedules**
  - **Celestial mapping**
  - **Mission planning & design**
- **Enable onboard revision & generation of these plans to respond to unexpected events and opportunities.**

JPL NASA UCLA